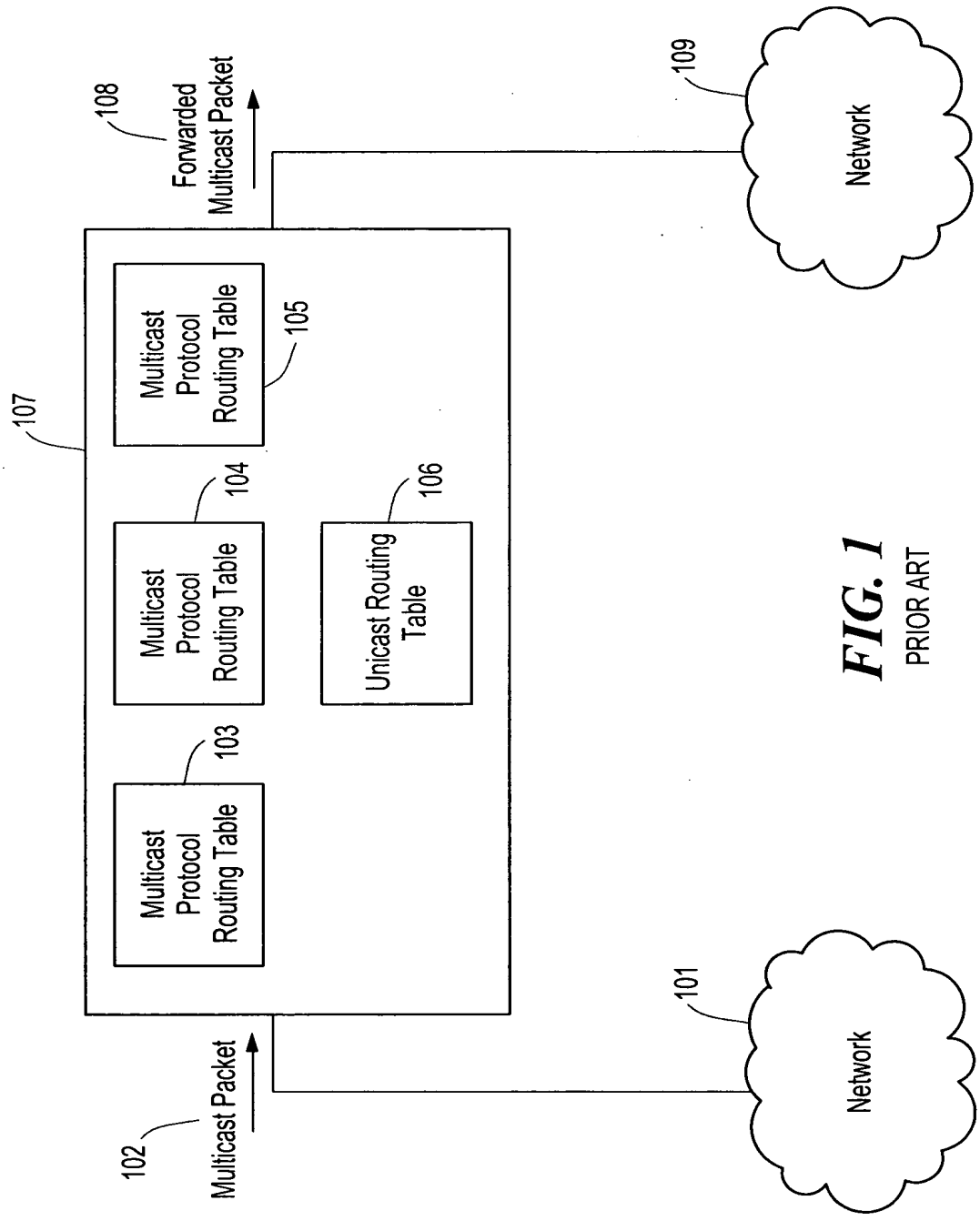

FIG. 1
PRIOR ART

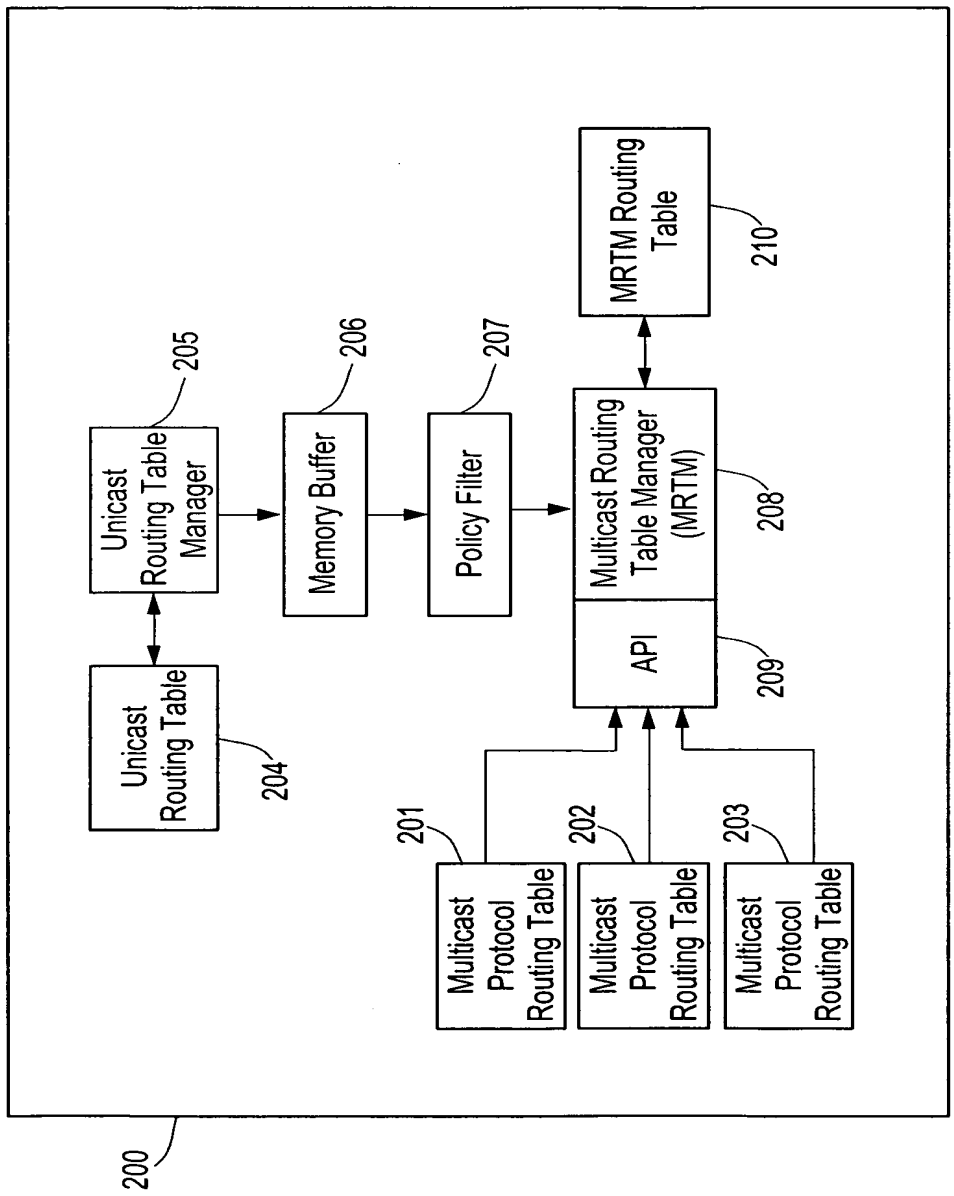
Unicast Routing Table Manager
205
Memory Buffer
206
Policy Filter
207
MRTM Routing Table
210
Multicast Routing Table Manager (MRTM)
208
API
209
Unicast Routing Table
204
Multicast Protocol Routing Table
201
Multicast Protocol Routing Table
202
Multicast Protocol Routing Table
203
200

FIG. 2

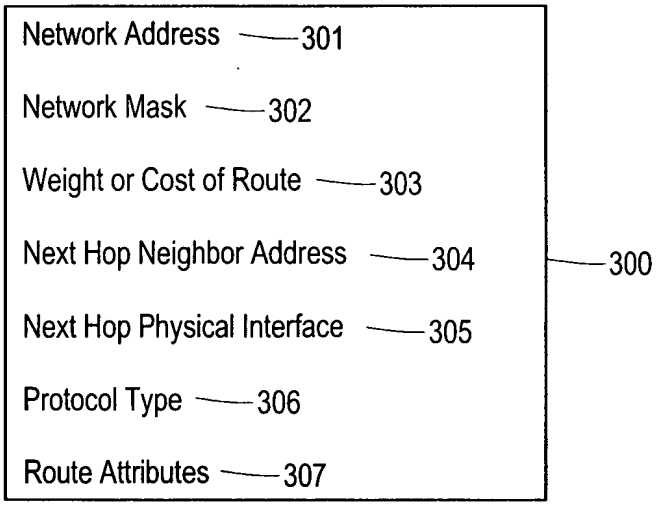
Network Address 301
Network Mask 302
Weight or Cost of Route 303
Next Hop Neighbor Address 304
Next Hop Physical Interface 305
Protocol Type 306
Route Attributes 307
300

FIG. 3

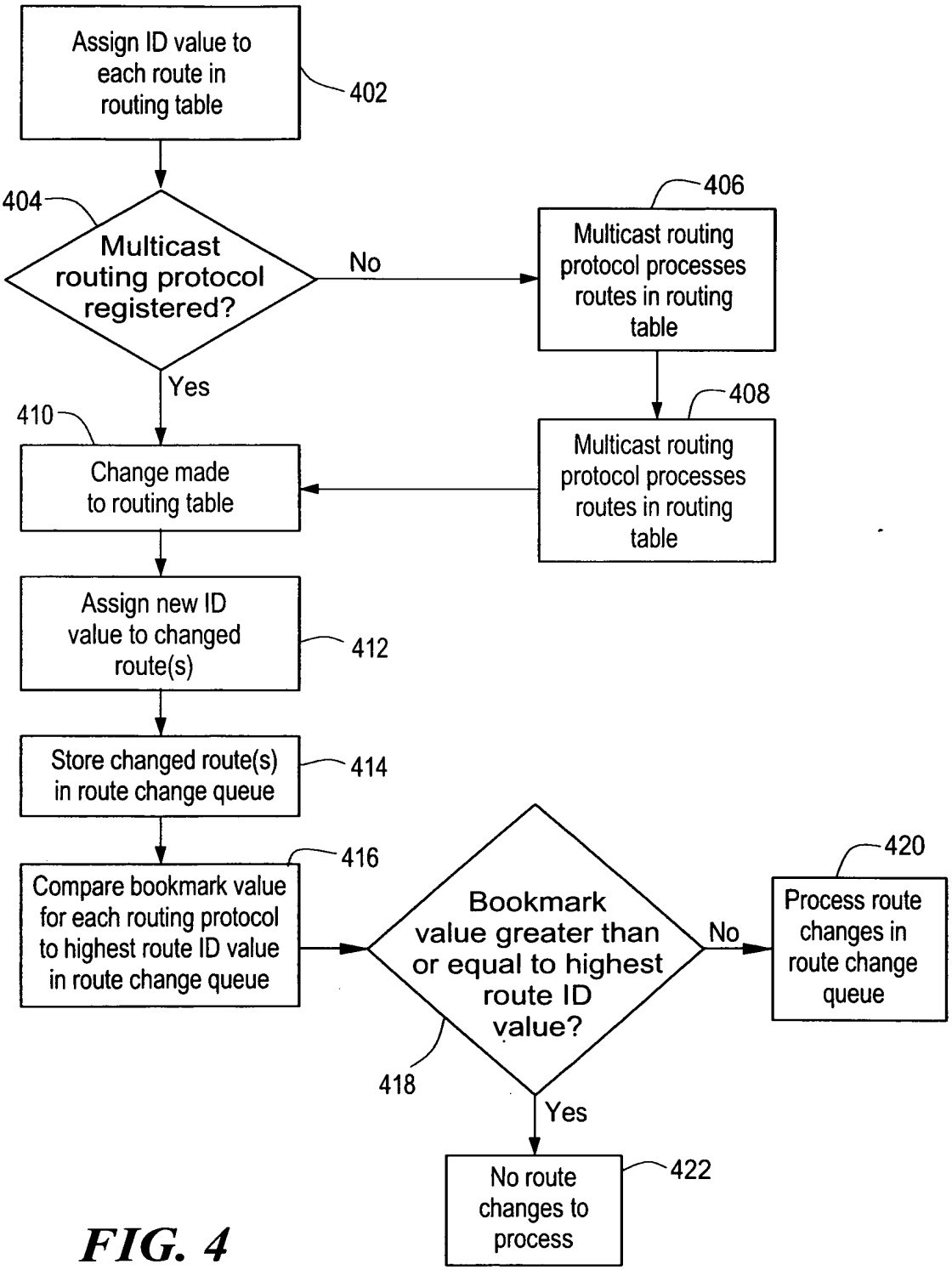
Assign ID value to each route in routing table
402
404
Multicast routing protocol registered?
No
406
Multicast routing protocol processes routes in routing table
Yes
408
410
Change made to routing table
Multicast routing protocol processes routes in routing table
Assign new ID value to changed route(s)
412
Store changed route(s) in route change queue
414
416
Compare bookmark value for each routing protocol to highest route ID value in route change queue
Bookmark value greater than or equal to highest route ID value?
No
420
Process route changes in route change queue
418
Yes
No route changes to process
422

FIG. 4

```
wfIpMrtmInjectRtTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF WfIpMrtmInjectRtEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "The Table of MRTM Inject Unicast routes Policy Rules"
        ::= { wfIpPolicyGroup 21 }

 wfIpMrtmInjectRtEntry OBJECT-TYPE
     SYNTAX  WfIpMrtmInjectRtEntry
     ACCESS  not-accessible
     STATUS  mandatory
     DESCRIPTION
             "An entry in the Mrtm Inject Route Rule Table"
     INDEX   { wfIpMrtmInjectRtIndex }
     ::= { wfIpMrtmInjectRtTable 1 }

 WfIpMrtmInjectRtEntry ::= SEQUENCE {
         wfIpMrtmInjectRtDelete
             INTEGER,
         wfIpMrtmInjectRtDisable
             INTEGER,
         wfIpMrtmInjectRtIndex
             INTEGER,
         wfIpMrtmInjectRtName
             DisplayString,
         wfIpMrtmInjectRtNetworks
             OCTET STRING,
         wfIpMrtmInjectRtAction
             INTEGER,
         wfIpMrtmInjectRtPreference
             INTEGER,
         wfIpMrtmInjectRtPrecedence
             INTEGER,
         wfIpMrtmInjectRtInject
             OCTET STRING,
         wfIpMrtmInjectRtInInterface
             OCTET STRING,
         wfIpMrtmInjectRtType
             INTEGER,
         wfIpMrtmInjectRtMetric
             INTEGER

   wfIpMrtmInjectRtDelete OBJECT-TYPE
        SYNTAX  INTEGER {
                   create (1),
                   delete (2)
                }
        ACCESS  read-write
        STATUS  mandatory
        DESCRIPTION
                "Create/Delete parameter."
        DEFVAL  { create }
       ::= { wfIpMrtmInjectRtEntry 1 }
```

FIG. 5A

```
wfIpMrtmInjectRtDisable OBJECT-TYPE
     SYNTAX  INTEGER {
                 enabled (1),

                 disabled (2)
             }
     ACCESS  read-write
     STATUS  mandatory
     DESCRIPTION
             "Enable/Disable parameter."
     DEFVAL  { enabled }
     ::= { wfIpMrtmInjectRtEntry 2 }

   wfIpMrtmInjectRtIndex OBJECT-TYPE
       SYNTAX  INTEGER
       ACCESS  read-only
       STATUS  mandatory
       DESCRIPTION
                   "Rule index number"
       ::= { wfIpMrtmInjectRtEntry 3 }

 wfIpMrtmInjectRtName OBJECT-TYPE
     SYNTAX  DisplayString
     ACCESS  read-write
     STATUS  mandatory
     DESCRIPTION
                 "Rule name - user specified name for this rule"
     ::= { wfIpMrtmInjectRtEntry 4 }

wfIpMrtmInjectRtNetworks OBJECT-TYPE
   SYNTAX  OCTET STRING
   ACCESS  read-write
   STATUS  mandatory
   DESCRIPTION
           "Network identification list. This identifies which
           networks will match this rule.  If non-null, the octet
           string contains one or more 3-tuples of this form:

            first octet:  exact (1) or range (2)
            next 4 octets:  network number
            next 4 octets:  network mask

           An entry with an 'exact' tag means to only match the
           specific network advertisement (number & mask).  An
           entry with a 'range' tag means to match any network
           number that falls in the range indicated by the number
           and mask.

           A null string also means 'match any route'."

   ::= { wfIpMrtmInjectRtEntry 5 }
```

FIG. 5B

```
    wfIpMrtmInjectRtAction OBJECT-TYPE
         SYNTAX  INTEGER {
                     accept (1),
                     ignore (3)
                 }
         ACCESS  read-write
         STATUS  mandatory
         DESCRIPTION
                 "action. 'accept' means that the route should be

                 imported from RTM to  the Mrtm routing table.  'ignore'
                 means don't consider the route"
         DEFVAL  { accept }
         ::= { wfIpMrtmInjectRtEntry 6 }


wfIpMrtmInjectRtPreference OBJECT-TYPE
    SYNTAX  INTEGER(0..16)
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
           "preference. This is a metric to be used to compare
           the preference path between inject route or the existing
           route in Mrtm routing table. If the injecting unicast
           route is preferred, then the value need to be set higher than
           the preference of the existing route.
           If the injecting unicast route path is preferred,
                         then the value need to be set greater than 0.
           This parameter only has meaning if the action is 'accept'."
    DEFVAL  { 1 }
    ::= { wfIpMrtmInjectRtEntry 7 }


wfIpMrtmInjectRtPrecedence OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
           "precedence. This is a metric to be used to compare
           this policy rule to the other rules that a given route may
           match.  A rule with a higher precedence value will be
           chosen over one with a smaller value.  In the case of
           a tie, the rule index is used (larger wins).

           Note that the policy match is not most specific
           so the precedence has to be used to select from
           multiple matches."
    ::= { wfIpMrtmInjectRtEntry 8 }
```

FIG. 5C

```
wfIpMrtmInjectRtInject OBJECT-TYPE
    SYNTAX  OCTET STRING
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
            "network injection list. this octet string should only be
            non-null if the action is 'accept' and if it is desired to
           insert networks into the routing table that differ from
            the actual advertised network.  For instance, if a number of
            networks in a certain range are learned, an aggregate
            advertisement could be inserted instead of the individual
            networks."

            If non-null, The octet string contains one 2-tuples of
            this form:

            first  4 octets: network number
            next   4 octets: network mask

            Upon receiving a route that matches this filter, the network
            in this list will be considered for the inclusion in the routing
            table.  If the list is null, the actual received network is

          considered."
    ::= { wfIpMrtmInjectRtEntry 9 }


wfIpMrtmInjectRtInInterface OBJECT-TYPE
    SYNTAX  OCTET STRING
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
            "Injected unicast routes inbound circuit list.
            This octet string contains one or more 4-octet IP addresses.
            If an interface address is included in this list, the unicast
            routes received on that interface match this rule will be
            accepted.

            If null, this filter applies to the unicast routes received on
            any interface."
    ::= { wfIpMrtmInjectRtEntry 10 }
```

FIG. 5D

```
wfIpMrtmInjectRtType OBJECT-TYPE
    SYNTAX  INTEGER  {
             static-route (1),
             rip (15),
             egp (16),
             ospf (17),
             bgp (18),
             direct-route (40),
             best-route (41),
             all-route (42)
            }
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
            "Select the injected route type from RTM. The value of each
             route type will be the same as unitcast route type. See
             define in ip_rt_types.h"
    DEFVAL   { best route }
    ::= { wfIpMrtmInjectRtEntry 11 )


  wfIpMrtmInjectRtMetric OBJECT-TYPE
    SYNTAX  INTEGER (1..31)
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
             "Route Metric. This value represents the cost of the external
             routes which are OSPF or unicast best route  to be injected
             into Mrtm routing table. The default value is set to 1."

    DEFVAL  { 1 }
    ::= { wfIpMrtmInjectRtEntry 12 }


 wfMrtm       OBJECT IDENTIFIER ::= { wfMrtmGroup 1 }


wfMrtmCreate  OBJECT-TYPE
    SYNTAX  INTEGER {
               created (1),
               deleted (2)
             }
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
             "Create/Delete a parameter. Default is created.
             Users perform a set operation on this
             object in order to create/delete MRTM table."
    DEFVAL  { created }
    ::= { wfMrtm 1 }
```

FIG. 5E

```
wfMrtmEnable  OBJECT-TYPE
    SYNTAX  INTEGER  {
            enabled (1),
            disabled (2)
            }
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
            "Enable/Disable parameter indicates whether
            this MRTM record is enabled or disabled."
    DEFVAL  { enabled }
    ::= { wfMrtm 2 }


wfMrtmState  OBJECT-TYPE
    SYNTAX  INTEGER  {
                up (1),
                down (2),
                init (3),
                notpres (4)
            }
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
            "The current state of the entire MRTM."
    DEFVAL  { notpres }
    ::= { wfMrtm 3 }


wfMrtmDebug  OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
            "This is a debug field for PGM.  Setting bits
            cause PGM to generate certain log messages.
            This field will NOT restart PGM.
            The follow bits maybe set in any combination
            (LS stands for least significant) :

            0x00000001 for no display
            0x00000002 for interface to RTM
            0x00000004 for interface to policy
            0x00000008 for interface to multicast protocols
            0x00000010 for route change or add or delete.
    ::= { wfMrtm 4 }
```

FIG. 5F

FIG. 5G

```
    wfMrtmHoldDown  OBJECT-TYPE
          SYNTAX  INTEGER (10..60)
          ACCESS  read-write
          STATUS  mandatory
          DESCRIPTION

           "This value specifies, in seconds, how long a route
           will be held in MRTM table after it becomes unreachable."
    DEFVAL   { 10 }
    ::= { wfMrtm 5 }


    wfMrtmFifoSize  OBJECT-TYPE
    SYNTAX  INTEGER (1..100)
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
           "This value represents the depth of the FIFO
           between RTM and MRTM used for the outstanding route changes.
           The memory will be pre-allocated as the size of
           x times 1000 of FIFO route entry."
    DEFVAL  { 5 }
    ::= { wfMrtm 6 }

    wfMrtmEstimatedNetworks  OBJECT-TYPE
         SYNTAX  INTEGER (10..200000)
         ACCESS  read-write
         STATUS  mandatory
         DESCRIPTION
                "This parameter indicates the estimated number of routes
                per slot that the router will need to keep in its routing
                table.This value is used for pre-allocating routing tables."
         ::= { wfMrtm 7 }

wfMrtmMaxRoutes  OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
           "Max number of routes, per slot. This is used to limit
           the size of routing tables. Note that routes are kept on a
           per-source network basis, independent of multicast group."
    ::= { wfMrtm 8 }

 wfMrtmActualRoutes  OBJECT-TYPE
     SYNTAX  INTEGER
     ACCESS  read-only
     STATUS  mandatory
     DESCRIPTION
           "Total actual entries currently in the routing table"
     ::= { wfMrtm 9 }
```